

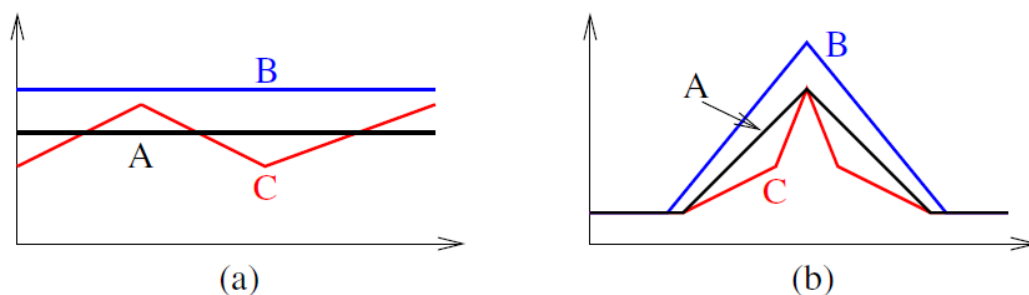
时间序列查询的两种思路：

- 1、匹配型，给定查询序列，在时间序列集合中找到和查询序列完全匹配的结果。这种查询的结果是结果较单一，只有完全匹配的才输出，差一个形状都不可以。
- 2、相似性，根据序列之间的距离来查询，距离越小则权值越高，这种查询则结果多样，但是部分结果的形状与给定序列不一致。

Keogh 对相似性查询做如下定义：

Indexing (Query by Content): **Given a query time series Q , and some similarity/dissimilarity measure $D(Q,C)$, find the nearest matching time series in database DB .**

但是相似其实有两种，一种是形状上的相似，另一种是度量上的相似。从 Keogh 的综述中看到大都关注量上的相似，而不关心形状。



欧氏距离是最简单的度量相似性的方法，但是对弯曲特别敏感，数据时间偏移或量上的缩放都会造成很大的度量值。DTW 和 LCSS 以及 EDR 能够减少弯曲敏感，但是它们在数据值上的又存在很大的敏感。如上图 a 中 A 和 B 在形状上要比 A 和 C 更相似，但是由于 B 的值更大，所以最后计算的距离值 A 和 B 更大，同样的在 b 图中，A 和 B 由于尺度不同导致二者的距离要比 A 和 C 的距离大。

一些表示方法开始关注两方面的因素，如 DDTW（是 DTW 的改造版）、LPAA（是对 PAA 的扩展）、DPLA（对 PLA 的变种）、SpADe（局部特征匹配的方法）、AMASS（用向量表示形状和尺度）等。

更一般地，国立新加坡大学 Yueguo Chen 提出时间序列有五个方面的特征，如下表：

	Time shifting	Time scaling	Amplitude shifting	Amplitude scaling	Noise
Euclidean		partially			
DTW	✓	✓			
CommonSub	✓	✓			✓
SpADe	✓	✓	✓	✓	✓

时间序列查询的两个重要问题：一是如何比较两个时间序列，二是如何构建索引。

能否先根据相似性查询得到尽量多样的结果，然后再根据形状匹配？

对于大规模的时间序列查询必须要建立索引，基于划分和基于符号是两类基本的序列表示方式，所以索引也有两类方法。划分的索引有 R-Tree、KD-Tree 等，而符号的索引有 B 树、前缀树等。

本周做了两个实验，都是符号匹配的，分别是 SDL 和 SAX 的。对 SDL 主要实验各种聚类算法（KMeans、IterativeKMeans、KMedoids 和 FarthestFirst 层次聚类），以及算法中的距离计算方法，有基于欧氏距离和 DTW。对 SAX 主要是实验了基于 KMeans 的聚类，其中的度量是基

于 SAX 自定义的度量规则。从结果上看，两类聚类都不是很理想，即使是使用 DTW 的 SDL 聚类也存在形状上的差异。

分析主要原因还是在于：

- 1、相同的度量距离，但是产生距离的时间点不同，如 A 和 B 与中心点都会产生距离，但是 B 在中间位置有一个峰值，但是 A 则在整个时间段内比较平滑；
- 2、存在中心点两侧到中心点的距离相近，但是其外形差异很大。如图中 C 和 D 就是分别处于中心点的两侧。

